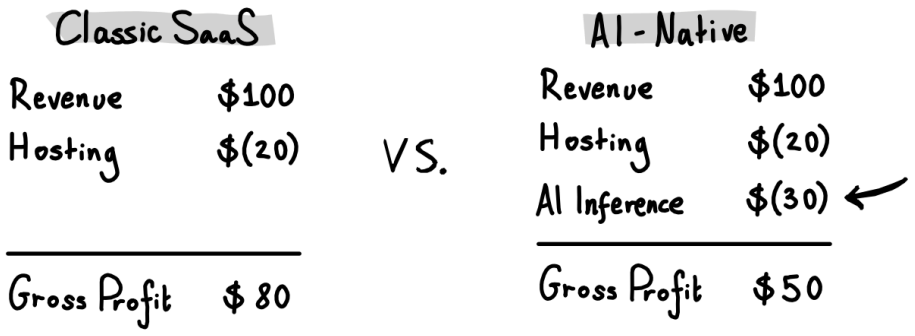
Something Is Changing in the Unit Economics of Software

N| nicolo.xyz/note/something-is-changing-in-the-unit-economics-of-software

Nicolo August 5, 2026



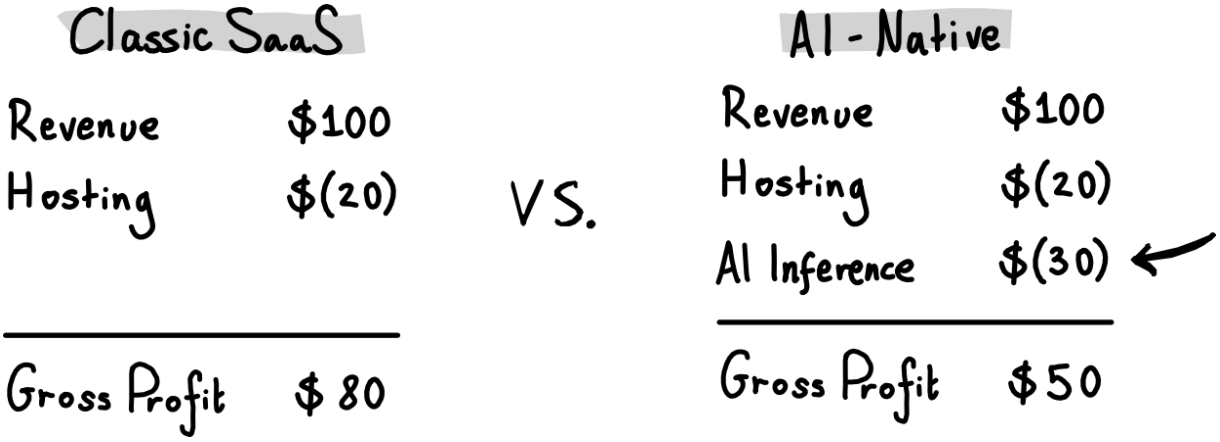
Software had a superpower. Build the product once, distribute it to a million users for roughly the same cost as distributing it to one. Every incremental customer flowed largely to the bottom line. This created a gross margin profile, 75-85%, that made software unlike any other industry in history.

Those margins made the unit economics of aggressive customer acquisition work in a way that seemed almost too good to be true. Once a customer was on the platform, they cost you almost nothing to serve. The lifetime value calculus was extraordinarily forgiving. That justified burning cash on growth, which produced the SaaS playbook that defined the last fifteen years: land customers aggressively now, because the unit economics get better as you scale, not worse. It was a genuine anomaly. Most industries don't work like that.

Then AI raised the stakes. Users began expecting something fundamentally different from software: not just tools that store and retrieve, but products that reason, generate, and respond. Meeting that expectation means making LLM calls, and LLM calls cost money.

The moment your product makes an LLM call every time a user does something, that superpower starts to erode. Every inference call[1] costs money. Every user interaction carries a compute cost that scales directly with usage. For the application

layer [2], this is a direct per-unit cost, real cash going out the door. Software always had a cost of goods sold: hosting, bandwidth, payment fees, though these were small, falling, and incidental to the product. AI makes them large, sticky, and inseparable from the core experience. The bill of materials finally matters.



Illustrative margin structure

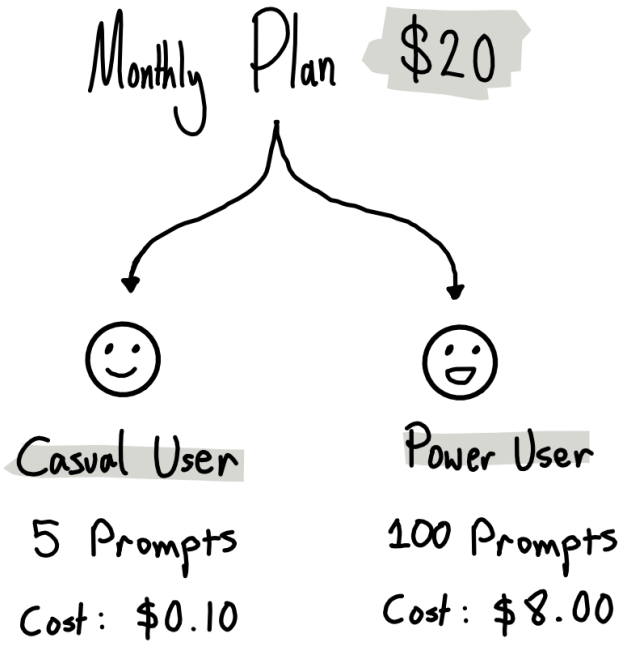
This introduces a tradeoff that did not exist in traditional SaaS. The model you choose to serve your users affects not just your P&L but your product quality, which affects your competitive position. Use a cheaper model and you protect margin but risk losing users to a competitor with a better experience. Use a frontier model[3] and you win on product but bleed on economics. For the first time, margin and quality are in direct conflict on a fundamental per-unit basis. It is a hardware-style decision in a software world: the kind of component selection and input cost management that manufacturers have always wrestled with, now sitting on the desk of a software founder for the first time.

In practice, most AI founders are well aware of this and take steps to manage it: running several models at once, routing everyday workloads to smaller or fine-tuned ones, escalating only the hard cases to frontier models. It is a rational response to a real constraint, but it does not make the tradeoff go away. As products scale, inference tends to become the dominant cost, displacing even talent as the primary expense. The tradeoff just becomes more central.

Right now, most companies are rationally ignoring the full implications of this. The market is moving fast, the land-grab is real, and burning on inference costs is a defensible short-term decision when competitive intensity is this high. It is also storing up a problem. At some point the market matures, the frenzy normalises, and you have to run a profitable business with a cost structure you have never been forced to think carefully about. The equilibrium will come. The founders who have been thinking about unit economics all along will be better positioned when it does.

If margins compress structurally, and they already are, the entire calculus collapses. Recent survey data from ICONIQ puts average gross margins on AI products at around 52% in 2026,[4] improving year on year as companies get smarter about cost management, but still well below the 75-85% that SaaS investors internalised as a baseline. You cannot justify burning aggressively on customer acquisition when those customers have real ongoing compute costs attached to them. The implicit contract of the SaaS era, grow now and margins will follow, stops holding when the cost structure does not improve with scale in the same way.

There is a second dimension to the squeeze, and it explains the pricing changes you can already see happening: AI does not just increase marginal cost, it increases variability in cost-to-serve in a way traditional SaaS never did. A power user making a hundred inference calls costs you dramatically more than a casual user making five. In traditional SaaS, both cost you almost nothing. Flat-rate subscriptions socialise that difference across your user base, which works until your heaviest users are expensive enough to blow up your margins. Usage-based pricing is the rational response: align what you charge to what it actually costs you to serve. It is not a coincidence that AI-native companies are steadily moving in this direction. The pricing model is changing because the cost model changed first.



The optimists will point out that inference costs are falling fast, and they are. You cannot, however, build a business model today on a cost breakthrough that has not happened yet. Even if costs do fall, Jevons paradox[5] tends to kick in: historically, when a resource gets cheaper, we find more uses for it rather than consuming less. Cheaper inference does not get banked as margin improvement; it gets consumed by

deeper integration, more calls per interaction, more autonomous agents running in the background. The cost per call drops but the calls per user multiply.

The playbook that emerges from all of this looks less like the SaaS era and more like how capital-efficient businesses have always been built. Unit economics from day one. Pricing architectures that reflect real cost structures. Growth funded by the business itself rather than subsidised by investors betting on margin expansion that may never arrive.

None of this is a eulogy for software. The founders who take these tradeoffs seriously, who think carefully about what they are running, for whom, and at what cost, are not building worse companies than the SaaS generation did. They might be building better ones. More considered, more resilient, more honest about where value is actually being created.

The zero marginal cost era produced extraordinary companies. It also produced a lot of businesses that mistook a favourable cost structure for a genuine competitive advantage, and never had to find out the difference. AI is forcing a harder conversation about margins, about pricing, about what it actually takes to build something sustainable. The rules of software are being rewritten. The best founders are already playing by the new ones.

1. An inference call is each time the model actually runs to produce an answer, as opposed to training, which is the one-off cost of building the model. ↩

2. The application layer is the vast majority of startups: those building on top of foundation models rather than owning the infrastructure themselves, such as Legora building on Anthropic or OpenAI. ↩

3. The largest, most capable, and most expensive tier of models available at any given time. ↩

4. ICONIQ, State of AI: Bi-Annual Snapshot, January 2026. Survey of ~300 executives at software companies building AI products. The 52% figure is the projected 2026 average gross margin on AI products, up from 41% in 2024. ↩

5. Named after William Stanley Jevons, who observed in 1865 that more efficient coal engines led to more coal consumption, not less. ↩

Opinions expressed are solely my own and do not express the views or opinions of my employer.